Modeling and Control of Manufacturing Systems

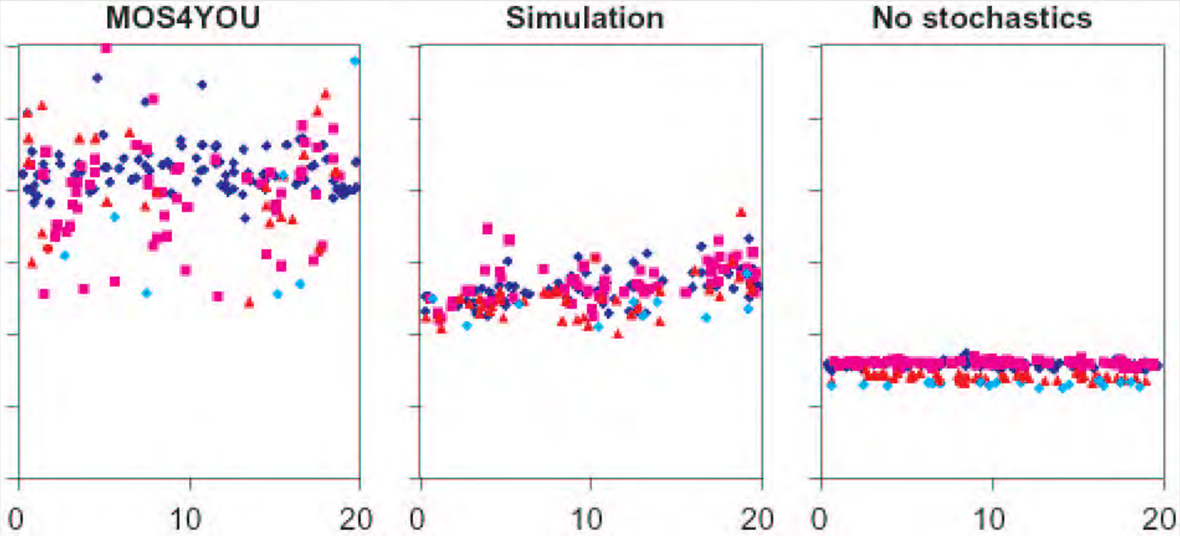
Erjen Lefeber

October 21, 2005

Discrete Event Modeling of a real factory

- raw process time t_0 and c_0
- setups t_s and c_s
- TBF t_f and c_f, TTR t_r and c_r
- operator delays
- rework
- …(!)

Results



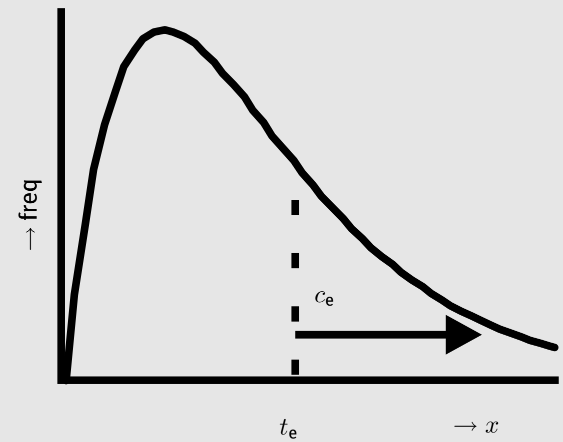
- smaller mean flow time
- smaller variance flow time

The effective process time method

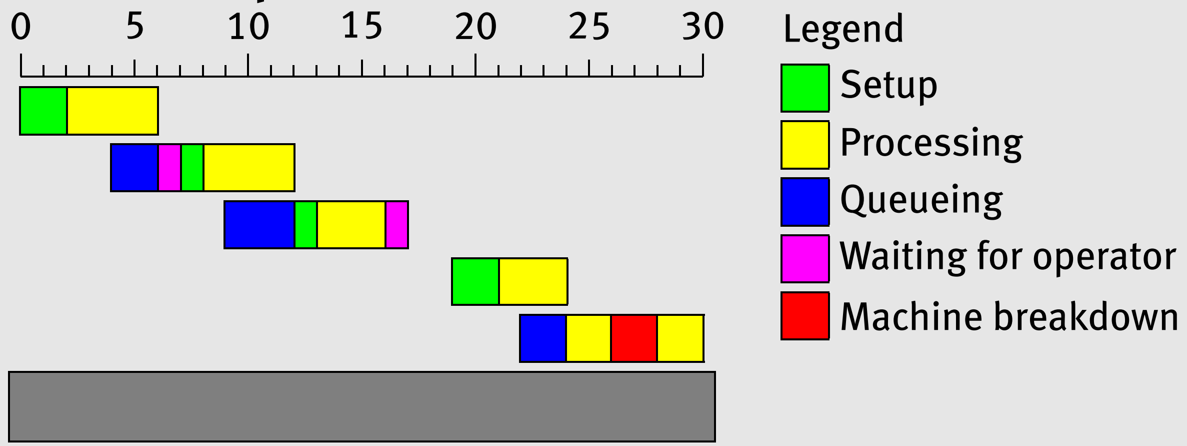
- raw process time t_0 and c_0
- setups t_s and c_s
- TBF t_f and c_f, TTR t_r and c_r
- operator delays
- rework
- ...(!)

Idea:

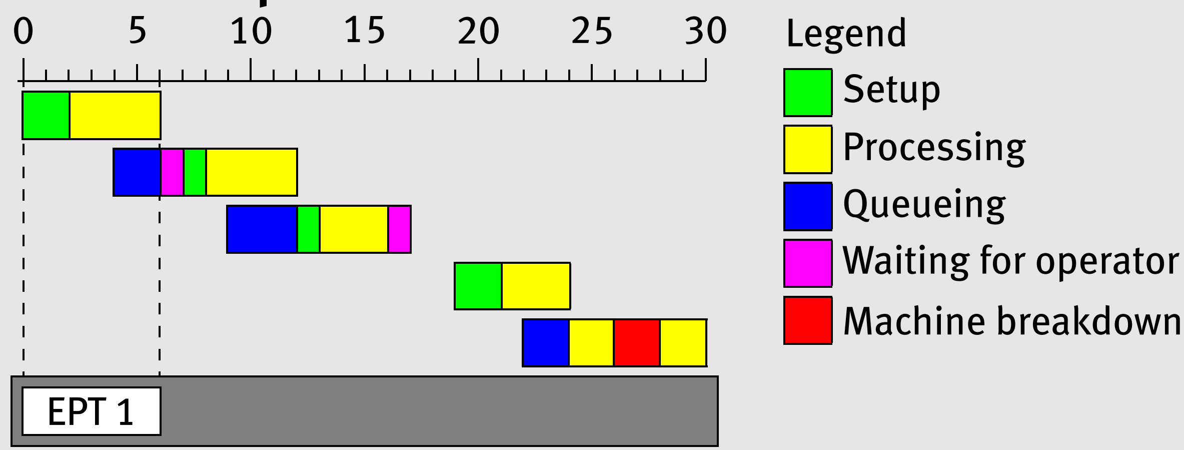
Combine all disturbances in one single EPT probability density function



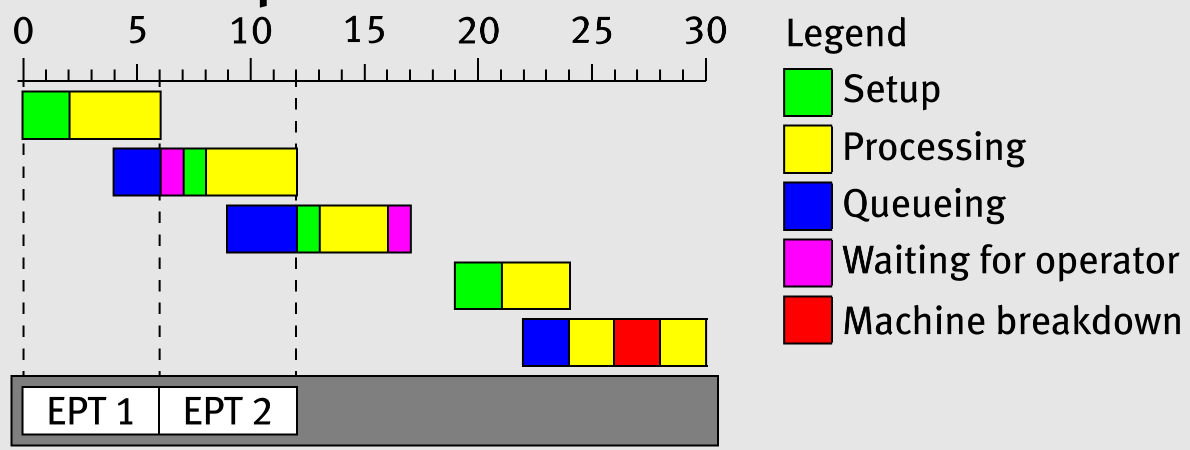
Effective process times



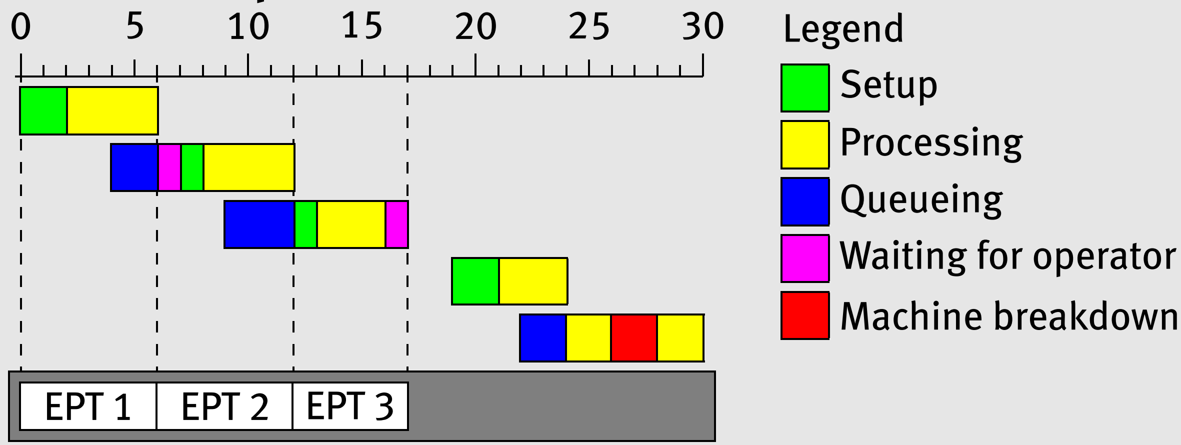
Effective process times



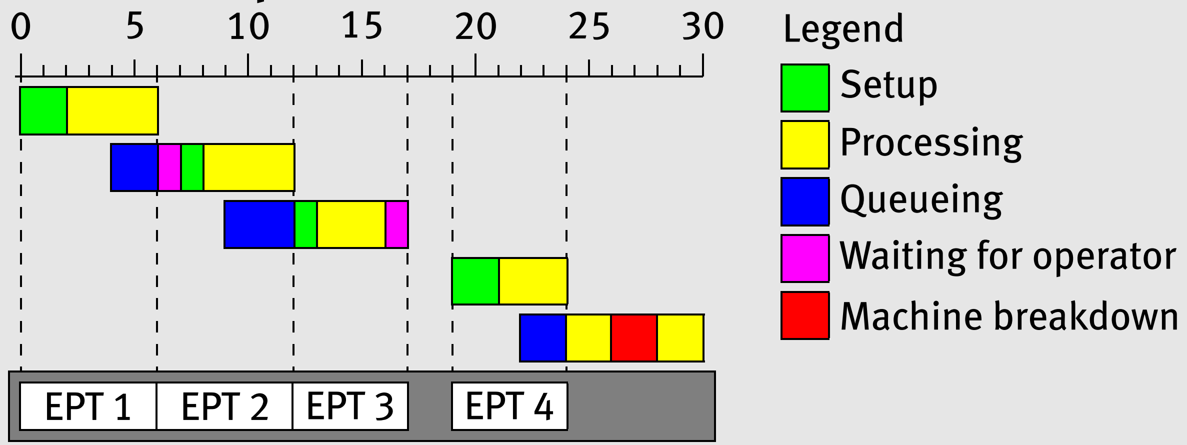
Effective process times



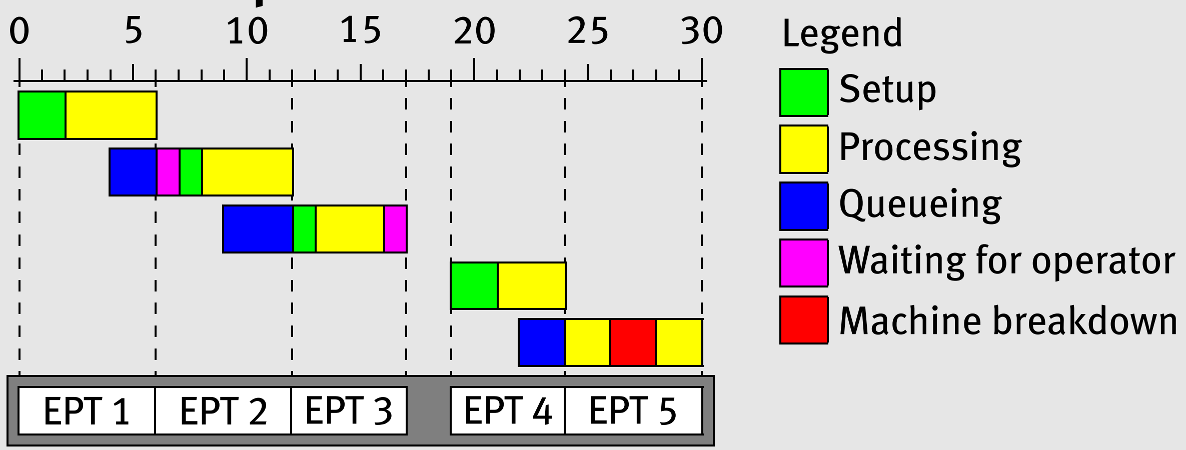
Effective process times



Effective process times



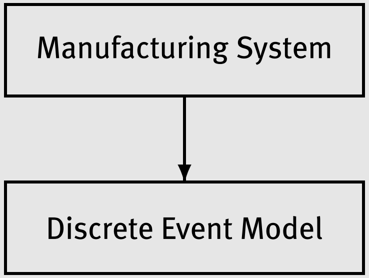
Effective process times



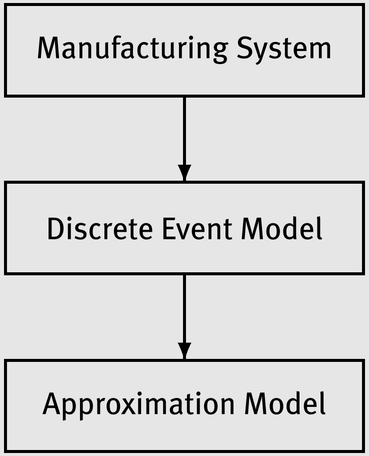
Control Framework

Manufacturing System

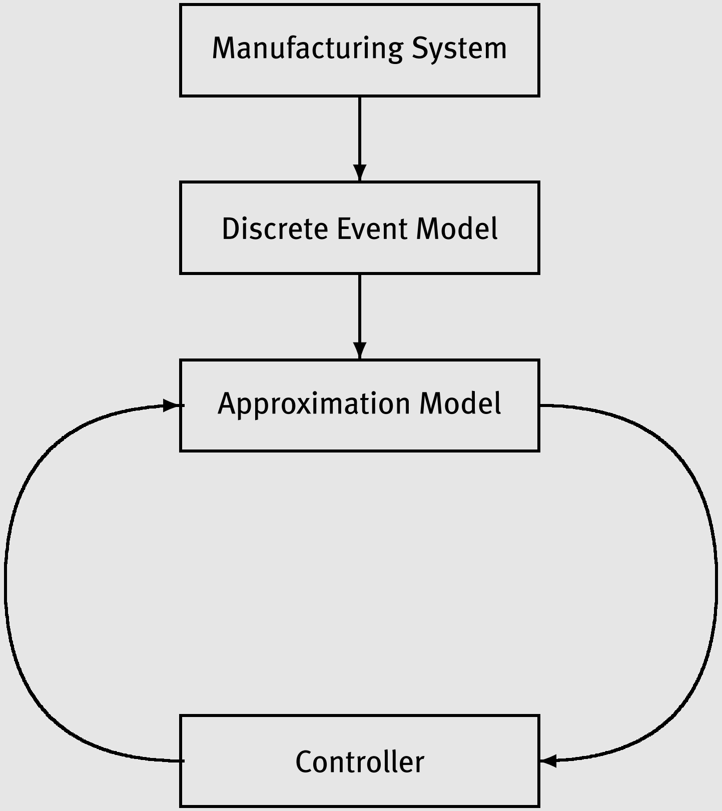
Control Framework



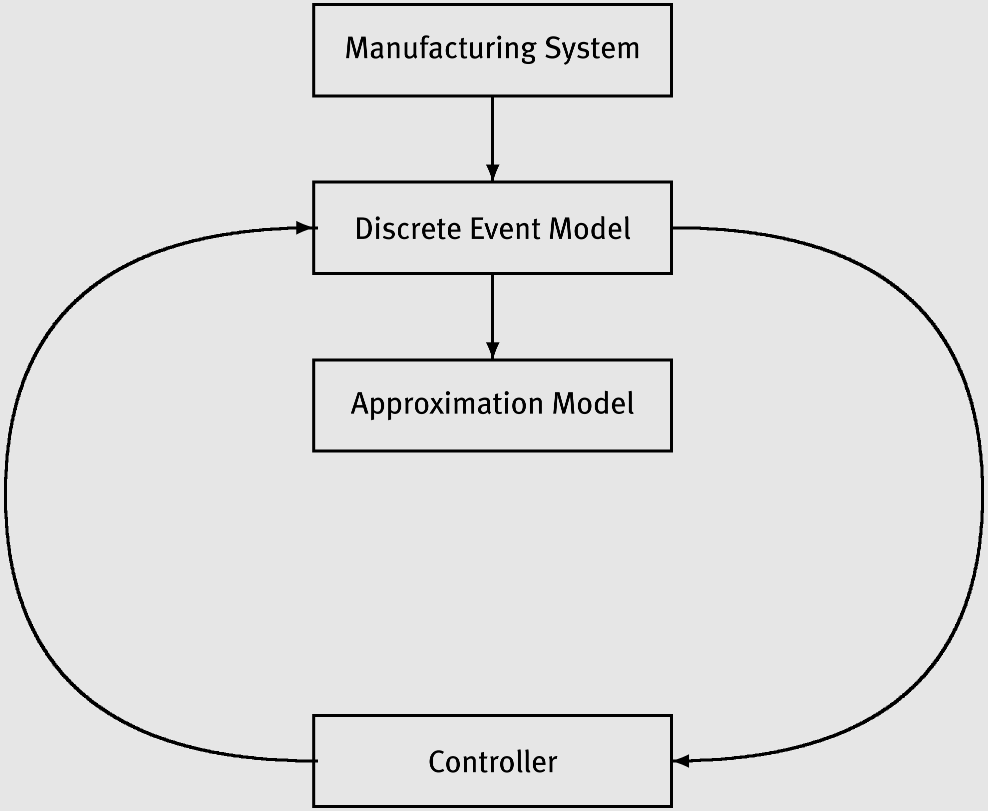
Control Framework



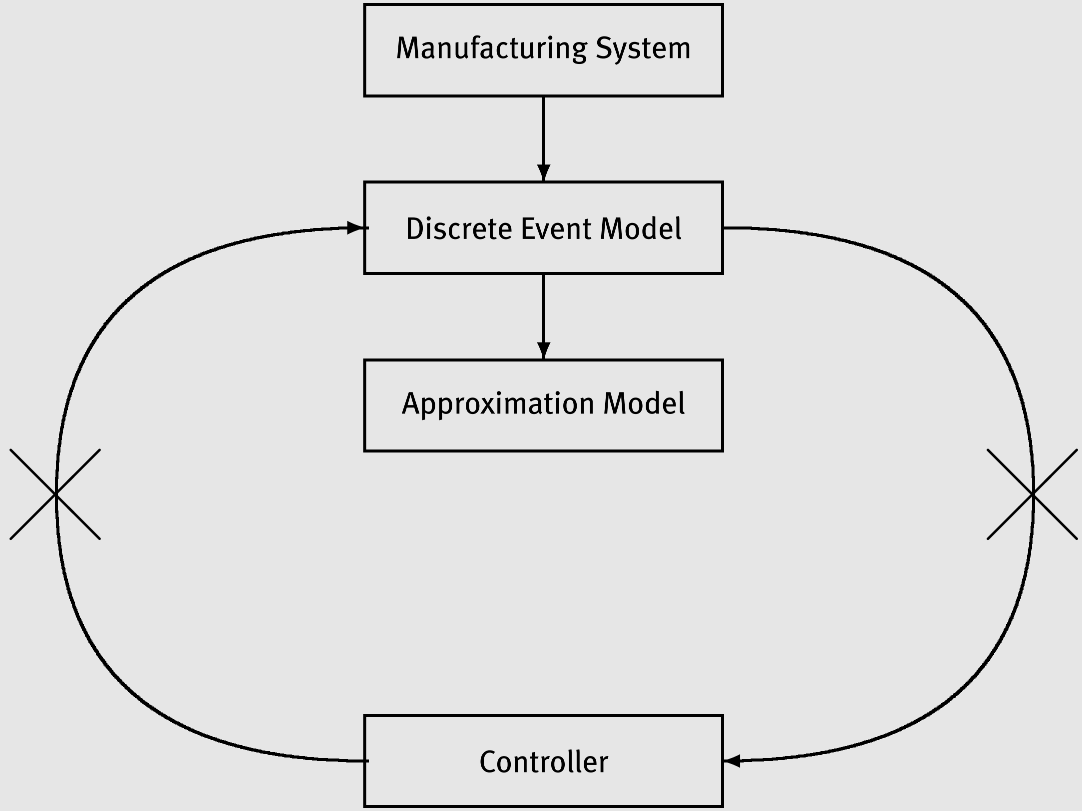
Control Framework



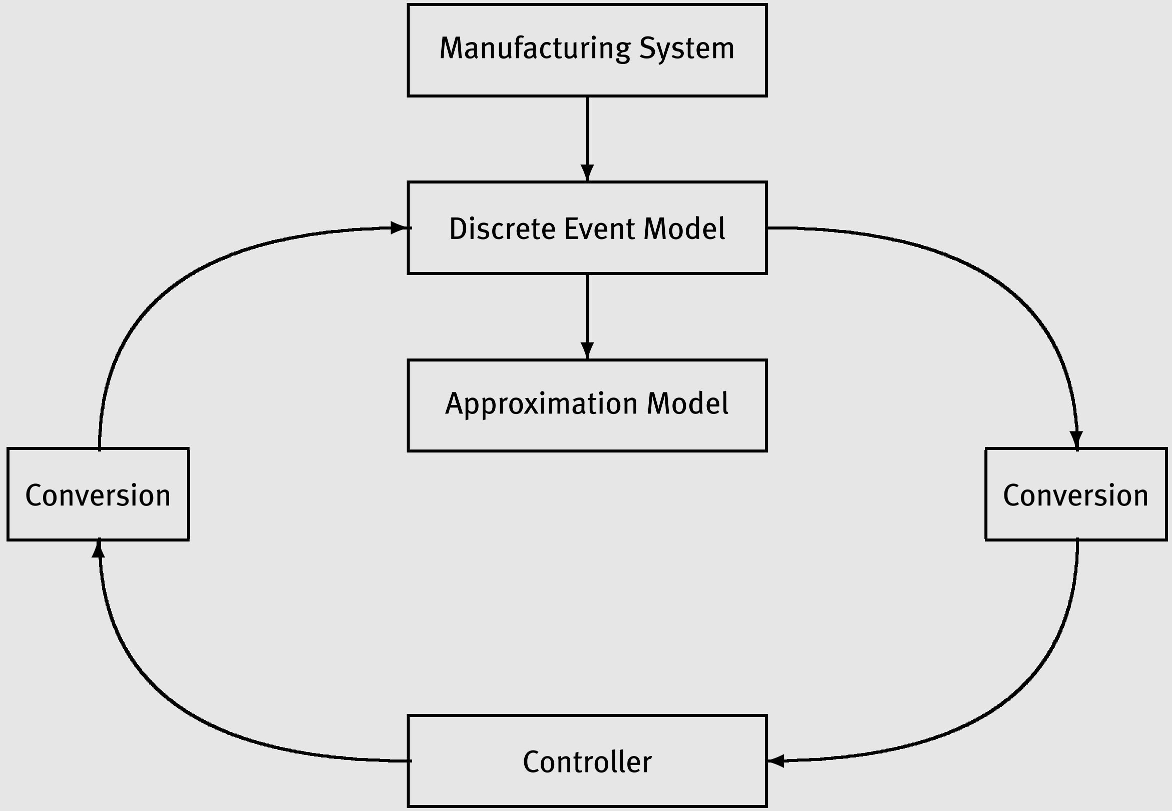
Control Framework



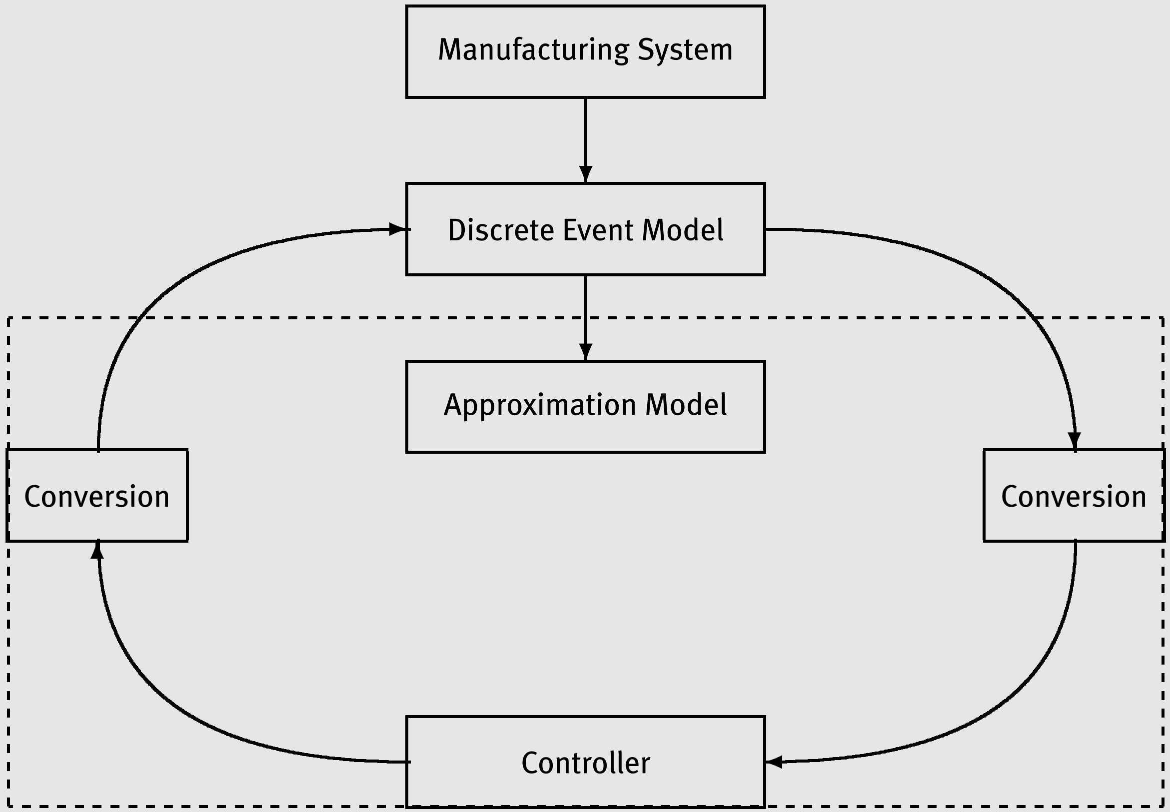
Control Framework



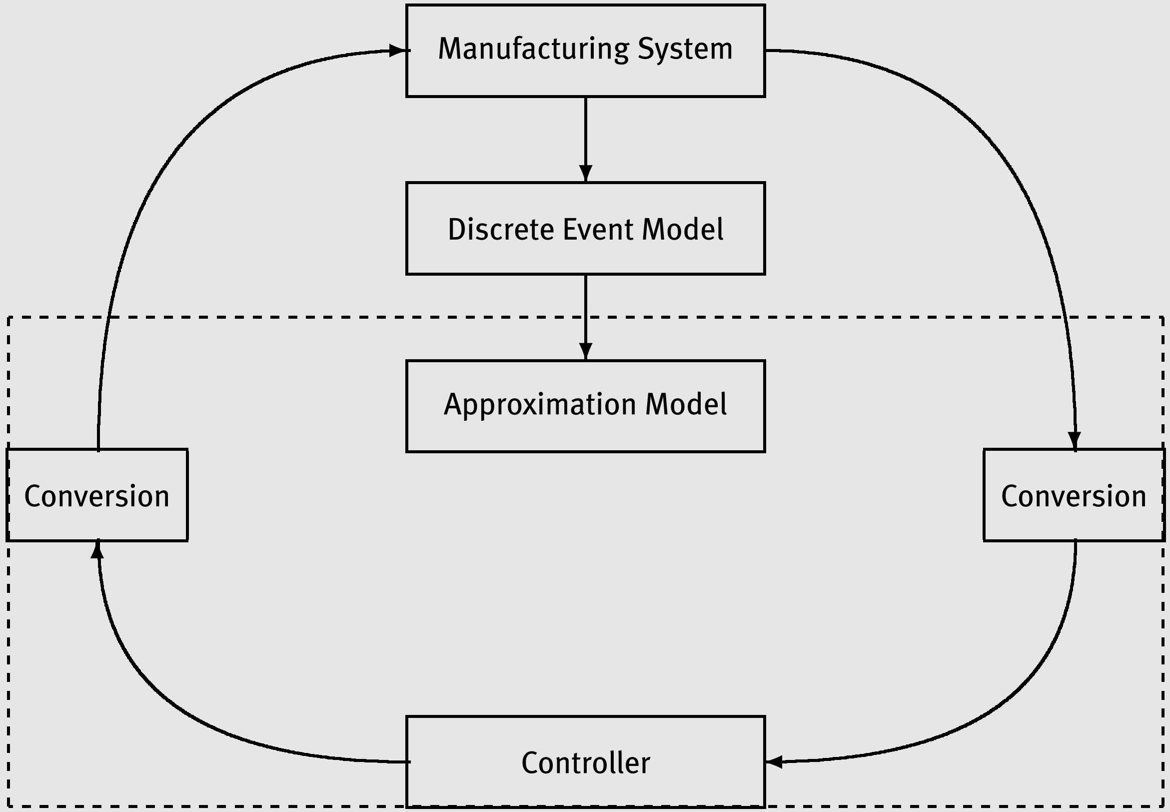
Control Framework



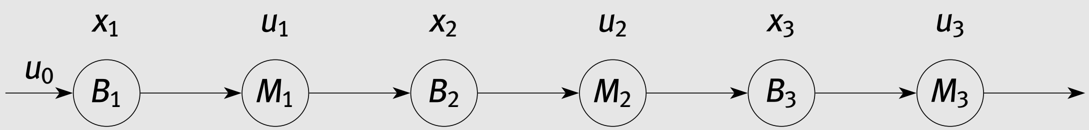
Control Framework



Control Framework



Approximation model



$$x_1(k+1) = x_1(k) + u_0(k) - u_1(k)$$

$$x_2(k+1) = x_2(k) + u_1(k) - u_2(k)$$

$$x_3(k+1) = x_3(k) + u_2(k) - u_3(k)$$

or

$$\begin{aligned} \dot{x}_1(t) &= u_0(t) - u_1(t) \\ \dot{x}_2(t) &= u_1(t) - u_2(t) \\ \dot{x}_3(t) &= u_2(t) - u_3(t) \end{aligned} \qquad \text{or} \qquad \begin{aligned} \dot{x}_1(t) &= u_0(t) - u_1(t) \\ \dot{x}_2(t) &= u_1(t - \tau_1) - u_2(t) \\ \dot{x}_3(t) &= u_2(t - \tau_2) - u_3(t) \end{aligned}$$

Modeling manufacturing flow

- density $\rho(x, t)$,
- speed $v(x, t)$,
- flow $u(x, t) = \rho(x, t)v(x, t)$,
- Conservation of mass: $\frac{\partial \rho}{\partial t}(x, t) + \frac{\partial \rho v}{\partial x}(x, t) = 0$.
- Boundary condition: $u(0, t) = \lambda(t)$

Modeling manufacturing flow

Armbruster, Marthaler, Ringhofer (2002):

- Single queue: $\frac{1}{v(x,t)} = \frac{1}{\mu}(1 + \int_0^1 \rho(s,t)\,\mathrm{d}s)$
- Single queue: $\frac{\partial \rho v}{\partial t}(x,t) + \frac{\partial \rho v^2}{\partial x}(x,t) = 0$

$$\rho v^2(0,t) = \frac{\mu \cdot \rho v(0,t)}{1 + \int_0^1 \rho(s,t)\,\mathrm{d}s}$$

- Re-entrant: $v(x,t) = v_0 \left(1 - \frac{\int_0^1 \rho(s,t)\,\mathrm{d}s}{W_{\max}}\right)$
- Re-entrant: $\frac{\partial \rho v}{\partial t}(x,t) + \frac{\partial \rho v^2}{\partial x}(x,t) = 0$

$$\rho v^2(0,t) = \rho v(0,t) \cdot v_0 \left(1 - \frac{\int_0^1 \rho(s,t)\,\mathrm{d}s}{W_{\max}}\right)$$

Lefeber (2003):

- Line of m identical queues: $v(x,t) = \frac{\mu}{m + \rho(x,t)}$

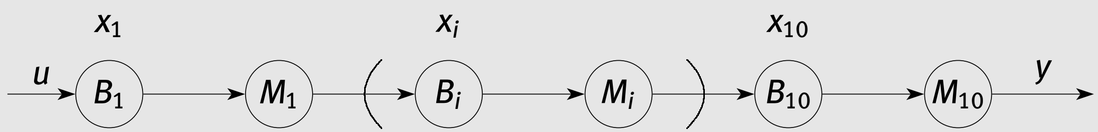
Validation studies

- Line of 15 identical machines
- Infinite queues
- FIFO-policy
- Exponential Effective Processing Times
- Step-response (initially empty, start rate λ)
- Model 1, 2, 5 versus averaged discrete event

Rampup to 50% utilization (averaged discrete event)

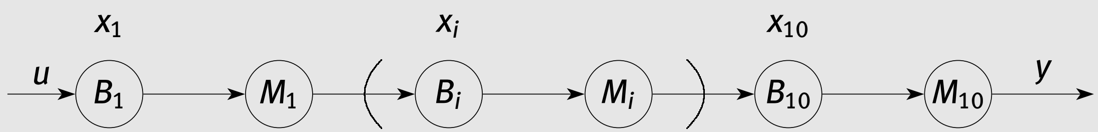
Rampup to 50% utilization (validation studies)

Control



- Number of machines $n = 10$
- Infinite queues
- FIFO-policy
- Exponential Effective Processing Times
- Mean processing time: 0.5h
- Desired $u = 0.75$ (1.5 lot per h)
- Initial WIP $x_i(0) = 0$

Nonlinear model



Discretizing PDE-model yields:

$$
\begin{aligned}
x_1(k+1) &= x_1(k) - \frac{\mu x_1(k)}{10 + x_1(k)} + u(k) \\
x_2(k+1) &= x_2(k) - \frac{\mu x_2(k)}{10 + x_2(k)} + \frac{\mu x_1(k)}{10 + x_1(k)} \\
&\vdots \\
x_{10}(k+1) &= x_{10}(k) - \frac{\mu x_{10}(k)}{10 + x_{10}(k)} + \frac{\mu x_9(k)}{10 + x_9(k)} \\
y(k) &= \frac{\mu x_{10}(k)}{10 + x_{10}(k)}
\end{aligned}
$$

MPC controller design

- Prediction horizon $p = 100\text{h}$
- Control horizon $p = 5\text{h}$
- Control constant over periods of 1h
- Time sampling: 40 steps per 1h

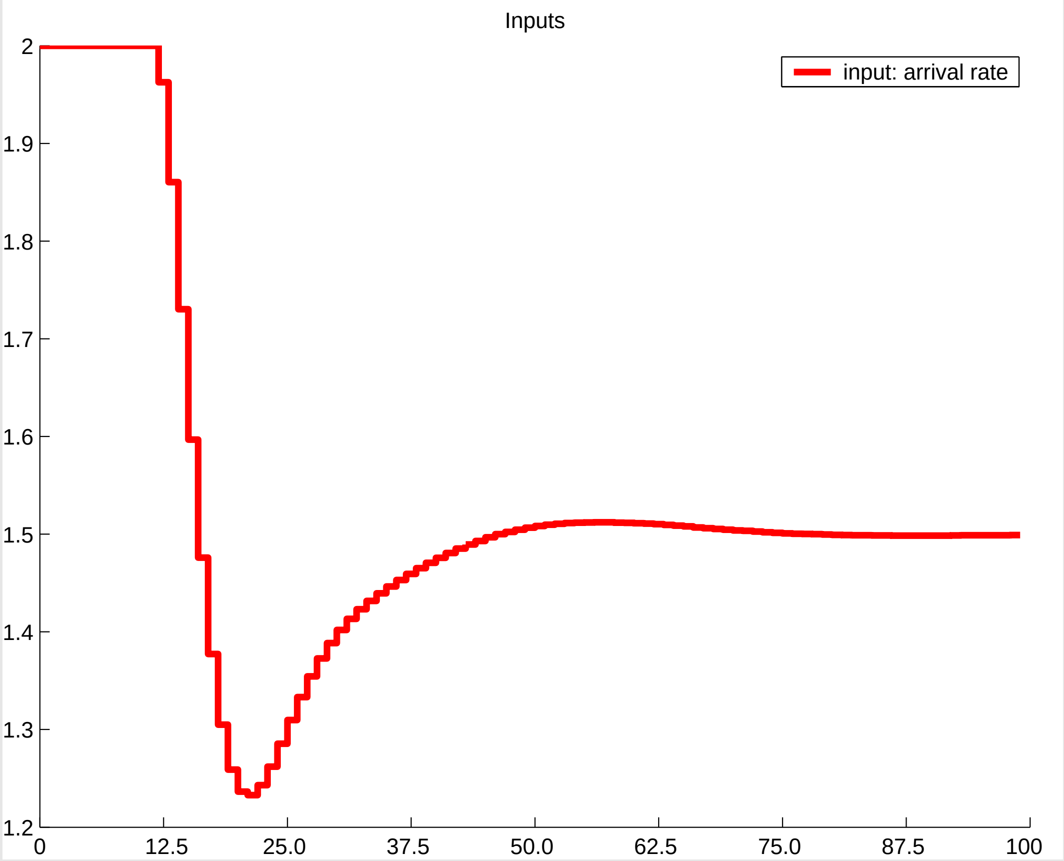
Cost function:

$$\min_u \sum_{i=0}^{p} ||y(k+i|k) - y_{\text{des}}||_Q^2$$

Constraints:

$$0 \leq u(k) \leq 2 \qquad 0 \leq \frac{\mu x_i(k)}{1 + x_i(k)} \leq 2$$

MPC based controller design



Conclusions

- EPT can be used to get from real data to simple queueing network model
- Control framework
- Control example